

Motion Planning for Global Localization in Non-Gaussian Belief Spaces

Saurav Agarwal

Amirhossein Tamjidi

Suman Chakravorty

Abstract—This paper presents a method for motion planning under uncertainty to deal with situations where ambiguous data associations result in a multimodal hypothesis on the robot state. In the global localization problem, sometimes referred to as the “lost or kidnapped robot problem”, given little to no a priori pose information, the localization algorithm should recover the correct pose of a mobile robot with respect to a global reference frame. We present a Receding Horizon approach, to plan actions that sequentially disambiguate a multimodal belief to achieve tight localization on the correct pose in finite time, i.e., converge to a unimodal belief. Experimental results are presented using a physical ground robot operating in an artificial maze-like environment. We demonstrate two runs wherein the robot is given no a priori information about its initial pose and the planner is tasked to localize the robot.

I. INTRODUCTION

In the domain of motion planning for mobile robots, situations may arise where data association between what is observed and the robot’s map leads to a multimodal hypothesis on the state, for example a kidnapped robot with no a priori information or a mobile robot operating in a symmetric environment (see Fig. 1). A large class of planning problems (e.g. visiting a fixed way point in space, driving through a narrow passage) require a well localized belief. State of the art belief space planning methods [2, 4, 6, 15, 19, 21, 25] rely on a Gaussian belief assumption to create solutions in the belief space. However, as discussed above, the Gaussian (unimodal) belief assumption may not always be a valid choice. This creates the requirement for a planner that can “actively” disambiguate a multimodal hypothesis.

We represent a multimodal hypothesis with a Gaussian Mixture Model (GMM) and use an Extended Kalman filter (EKF) based Multi-Hypothesis Tracking (MHT) approach to propagate the belief. Our Multi-Modal Motion Planner (M3P) achieves disambiguation in a multimodal belief by first finding a neighboring location (referred to as target state) for each belief mode and then creating a candidate action to guide the belief mode to its target state such that these actions lead to information gathering behavior. The target states are chosen such that different modes of the robot’s belief are expected to observe distinctive information, thus accepting or rejecting hypotheses in the belief.

Contributions: The main contributions of this work can be summarized as follows:

- 1) We develop a novel method for picking target states and creating candidate trajectories for a multimodal belief and choosing the best one, such that the maximum disambiguating information is observed which helps in rejecting incorrect hypotheses.
- 2) We prove that under certain realistic assumptions, through a process of iterative hypothesis elimination, our method can localize to the true robot pose.
- 3) We demonstrate our method on a 2D navigation problem in which a robot begins in a kidnapped situation and recovers its pose.

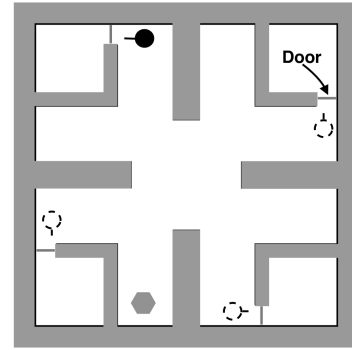


Fig. 1: A simple scenario depicting a multi-hypothesis localization problem, in a world with 4 rooms with identical doors. The true hypothesis is depicted by the solid disk, whereas the others are depicted by dashed disks. As the robot cannot distinguish between the doors, all hypotheses are equally likely.

Albeit the scenario used to motivate the problem is the kidnapped robot situation, the method proposed is general, and can be extended to any planning situation where a multimodal belief arises in the robot state due to ambiguous data associations (a common practical issue in robot localization [24]). In the proceeding section, we present relevant related work, and discuss how our approach compares with them. In Section III we state some preliminaries followed by the problem formulation. In Section IV we present our method followed by experimental results in Section V.

II. RELATED WORK

Recent work in sampling-based methods for belief space planning have shown promising results. The fundamental goal being to plan actions that minimize uncertainty such that a

mobile robot can localize accurately to act safely and reliably. Methods such as [4, 6, 15, 19, 21, 25] provide solutions that depend on the initial belief. Recent developments in [1, 2] extend belief space planning to multi-query settings (cases where multiple planning requests are made sequentially) by creating a belief space variant of a Probabilistic RoadMap (PRM) [14]. We note that all the methods mentioned above rely on a Gaussian belief assumption. Additionally, the aforementioned methods assume that the data associations between observations and information sources (e.g., landmarks) are known and unambiguous. In contrast, this work does not assume that the data associations are unambiguous or that the belief is unimodal. Another class of methods is the trajectory optimization approach which can be implanted in a Receding Horizon Control (RHC) framework for planning. A widely used approach in RHC-based control is to approximate the stochastic system with a deterministic one by substituting the random variables with their most-likely values [3]. Methods such as [5, 11, 19], as well as the work in this paper assume the most-likely values for the unknown future observations in the planning stage.

Recent work in [18, 20] extends belief space planning to non-Gaussian beliefs. The authors investigate a grasping problem with a multimodal hypothesis on the gripper's state. Their method picks the most-likely hypothesis and a fixed number of samples from the belief distribution, then using an RHC approach, belief space trajectories are found that maximize the observation gap between the most-likely hypothesis and the drawn samples, which helps to accept or reject the most-likely hypothesis. The method in [17] builds upon the work in [18] wherein the author transposes the non-convex trajectory planning problem in belief space to a convex problem. Compared to [17, 18, 20], our method is better suited to deal with more severe cases of non-Gaussian belief space planning such as the kidnapped robot scenario. Such scenarios may not be possible to address using the trajectory optimization based techniques of [17, 18] in their current form, due to the difficulty of generating an initial feasible plan for the widely separated modes in the presence of obstacles (see Fig. 1 for an example of widely separated modes).

In the domain of global localization with a priori maps, [7] showed that finding the optimal (shortest) plan to re-localize a kidnapped robot with multiple hypothesis in a deterministic setting (no sensing or motion uncertainty) is NP-hard. At best a greedy localization strategy can be developed whose plan length is upper bounded by $(n-1)d$, where n is the number of hypothesis and d is the length of the optimal plan. Compared to [7], we do not assume perfect sensing or actuation. In [8], the authors develop an active localization method in a grid based scheme for a known map. Their planning method considers arbitrary targets in the robot's local coordinate frame as atomic actions (e.g., move 1m right and 4m forward). The optimal action is selected based on the path cost and the expected decrease in entropy at the target. Compared to [8], our target selection methodology (Section IV-A and IV-B1) is active, i.e., M3P uses the a priori map information to select targets such

that by visiting them, observation gap between belief modes is maximized resulting in successive disambiguation.

Successful application of the Gaussian mixture model to multi-hypothesis tracking for robot localization was shown in [12, 22, 23]. In [12], the authors present a greedy heuristic-based planning strategy to disambiguate a multimodal hypothesis along with an experimental demonstration for a kidnapped robot. The paper alludes to the fact that planning can be improved with a POMDP (Partially Observable Markov Decision Process) style approach. The method of [10] uses a hybrid approach in which a particle filter is used for hypothesis generation and an EKF is used to track each hypothesis, safe trajectories are planned by picking a point in the vicinity of obstacles to disambiguate the hypothesis. Compared to [10, 12], we present a planning approach that explicitly reasons about the belief evolution as a result of actions in the planning stage and picks an optimal policy from a set of candidates. We now proceed to formally describe our problem statement.

III. PROBLEM STATEMENT

Let x_k , u_k , and z_k represent the system state, control input, and observation at time step k respectively. Let \mathbb{X} , \mathbb{U} , and \mathbb{Z} denote the state, control, and observation spaces respectively. It should be noted that in our work, the state x_k refers to the state of the mobile robot, i.e., we do not model the environment and obstacles in it as part of the state. The sequence of observations and actions are represented as $z_{i:j} = \{z_i, z_{i+1}, \dots, z_j\}$ and $u_{i:j} = \{u_i, u_{i+1}, \dots, u_j\}$ respectively. The non-linear state evolution model f and measurement model h are denoted as $x_{k+1} = f(x_k, u_k, w_k)$ and $z_k = h(x_k, v_k)$, where $w_k \sim \mathcal{N}(0, Q_k)$ and $v_k \sim \mathcal{N}(0, R_k)$ are zero-mean Gaussian process and measurement noise, respectively.

The belief b_k at time t_k can be represented by a Gaussian Mixture Model (GMM) as a weighted linear summation over Gaussian densities. Let $w_{i,k}$, $\mu_{i,k}$ and $\Sigma_{i,k}$ be the weight, mean vector, and covariance matrix associated to the i^{th} Gaussian $m_{i,k}$ respectively at time t_k , then

$$b_k = \sum_{i=1}^{M_k} w_{i,k} m_{i,k}, \quad m_{i,k} \sim \mathcal{N}(\mu_{i,k}, \Sigma_{i,k}), \quad (1)$$

where M_k is the number of modes at time t_k . We state our problem as follows:

Given an a priori map, the system dynamics and observation models, construct a belief space planner $G(b_k)$ such that under the planner G , given any initial multimodal belief b_0 , the belief state process evolves such that $M_k = 1$, for some finite time steps k .

Note that in certain cases, the map may not allow actions that lead to hypothesis elimination such that the belief converges to a unimodal distribution (e.g., in a map with two identical closed rooms, if a robot is kidnapped and placed in either room, at best the robot can assign some probability to being in each room based on its observations). In such cases, M3P attempts to minimize M_k (by design). Moreover,

note that it is not possible to pre-compute what this minimum value of M_k is without knowing the true hypothesis in the multimodal belief.

IV. METHODOLOGY

We begin by defining certain key concepts used in the M3P planner.

Uniqueness Graph: A graph U_g , whose nodes are states sampled from the collision free space and whose edges relate the similarity of information observed at the sampled locations.

Target State: A target state $v_i^{tt} \in U_g$ for mode m_i is a node of the uniqueness graph which belongs to some neighborhood of radius R of the mode's mean μ_i such that if each mode were to visit its target, the observations at the target would lead to disambiguation in the belief.

Candidate Policy: A candidate policy π_i for mode m_i is a local feedback controller that guides the mode to its target v_i^{tt} .

The M3P methodology has two phases, an offline phase in which we generate U_g and an online phase in which we use the offline computations and plan in a receding horizon manner to disambiguate the belief.

A. Computing the Uniqueness Graph: Offline Phase

The uniqueness graph U_g is constructed by uniformly sampling the configuration space and adding these samples as nodes of U_g . Once a node is added, we simulate the observation for the state represented by that node. Let v_α be one such node and z^{v_α} be the observation if the robot were to be in state v_α . We add an edge $E_{\alpha\beta}$ (undirected) between two nodes v_α and v_β if the simulated observations at both nodes are similar. Further, the edges are weighted and the weight is dependent on the degree of similarity in the information observed. For example, in our landmark based observation model, each landmark has a signature (appearance), thus if the state at v_α observes $z^{v_\alpha} = \{s_1, s_2, s_3\}$, i.e., the landmarks with signature s_1, s_2 and s_3 and at v_β observes $z^{v_\beta} = \{s_1, s_2, s_4\}$, the edge $E_{\alpha\beta}$ would be given a weight of 2 as there two similar landmarks observed ($z^{v_\alpha} \cap z^{v_\beta} = \{s_1, s_2\}$). A higher edge weight signifies that the states represented by the vertices of that edge are more likely to observe similar information. The lack of an edge between two nodes means that if a robot were to make an observation at those two states, it would see distinctly different information. The complexity for the construction of U_g is $\mathcal{O}(n^2)$ (where n is the number of samples) as each sample (node) is checked with every other for information overlap.

Issues: Due to its random nature, sampling may often occur in regions of low information density (e.g., regions where there are few or no landmarks). One can often circumvent this issue by increasing the number of samples. As U_g is computed offline, the online performance is not significantly affected. Recent work in [16] suggests a localization aware sampling strategy which may be explored in future work.

B. RHC based Planning: Online Phase

In a multimodal scenario, we claim that the best action to take is one that guides a robot without collision through a path that results in information gain such that a disambiguation occurs (one or more hypotheses are rejected, see Fig. 2). Algorithm 1 describes the online planning process. In step 3, the planner picks target states for each belief mode such that visiting a target can either prove or disprove the hypothesis. In step 4, the planner generates a set of candidate policies to drive each mode to its target. In step 5, the expected information gain for each policy is computed and we pick the best one, and in step 7, the multimodal belief is propagated according to the action and observations. We proceed to describe steps 3, 4, 5 and 7 of Algorithm 1 below.

Algorithm 1: M3P: MultiModal Motion Planner

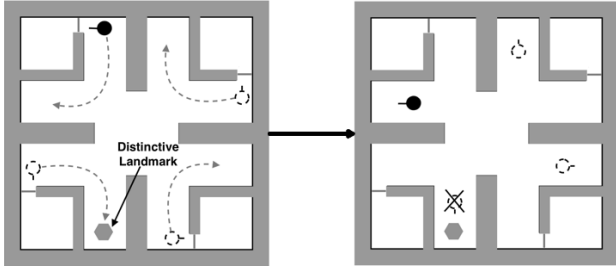
```

1 Input:  $b$ 
2 while  $b \neq \mathcal{N}(\mu, \Sigma)$  do
3    $\{v^{tt}\} \leftarrow$  Pick target states for belief modes (see Alg. 2);
4    $\Pi \leftarrow$  Generate candidate policies to connect each mode to its target;
5    $\pi^* \leftarrow$  Pick policy from  $\Pi$  with maximum expected information gain;
6   forall the  $u \in \pi^*$  do
7      $b \leftarrow$  Apply action  $u$  and update belief (see Alg. 3 for weight update calculation);
8     if Change in number of modes or Expect a belief mode to violate constraints then
9       break;
10 return  $b$ ;
```

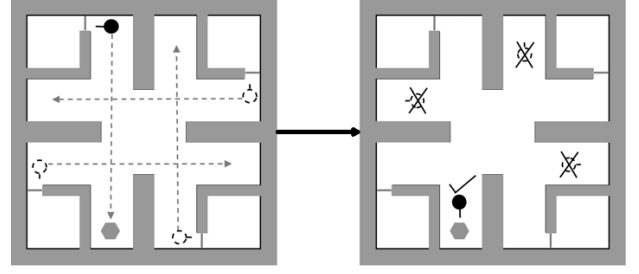
1) *Picking the target state for a mode:* Let us pick a mode $m_{i,k} \sim \mathcal{N}(\mu_{i,k}, \Sigma_{i,k})$ from the belief. To find the target for $m_{i,k}$, we first choose the set of nodes $N_{i,k} \in U_g$ (Section IV-A) which belong to the neighborhood of the mean $\mu_{i,k}$ at time t_k . Then, we find the target node $v_{i,k}^{tt} \in N_{i,k}$ which observes information that is least similar in appearance to that observed by nodes in the neighborhood $N_{j,k}$ of mode $m_{j,k}$ where $j \neq i$. We are trying to solve the optimization problem,

$$\{v_{1,k}^{tt}, v_{2,k}^{tt}, \dots, v_{M_k,k}^{tt}\} = \arg \min_{v_{i,k} \in N_{i,k}} \bigcap_{i=1}^{M_k} z^{v_{i,k}}, \quad (2)$$

where $z^{v_{i,k}} = h(v_{i,k}, 0)$. Thus, Eq. 2 solves for the set of target nodes, such that if each mode $m_{i,k}$ were to visit its target node $v_{i,k}^{tt}$, then we can prove or disprove the i -th mode. To solve Eq. 2, first we calculate the total weight of the outgoing edges from every node $v_{i,k} \in N_{i,k}$ to nodes in all other neighborhoods $N_{j,k}$ where $j \neq i$. The node which has the smallest outgoing edge weight, is the target candidate $v_{i,k}^{tt}$ for $m_{i,k}$ as the observation $z^{v_{i,k}^{tt}}$ would be least similar to the information observed in the neighborhood of all other



(a) Candidate A leads to negative information for the mode in lower left corner. It expects to see the distinctive landmark which robot doesn't observe, and is thus rejected.



(b) Candidate B leads the true hypothesis to be confirmed as the robot sees the distinctive landmark.

Fig. 2: Extending the example in Fig. 1, we depict how M3P creates candidate trajectories and picks the optimal one. For clarity we show only two candidates A & B and the effect of their execution. Candidate B results in complete disambiguation and is clearly a better choice.

modes m_j where $j \neq i$. Algorithm 2 describes in detail the steps involved.

Algorithm 2: Finding the target for i -th mode

```

1 Input:  $b_k, i, U_g$ 
2 Output:  $v_{i,k}^{tt}$ 
3 forall the  $l \in [1, M_k]$  do
4    $N_{l,k} \leftarrow$  Find nodes in  $U_g$  within neighborhood of
    $\mu_{l,k}$ ;
5  $minWeight \leftarrow$  Arbitrarily large value;
6  $v_{i,k}^{tt} \leftarrow -1$ ;
7 forall the  $v \in N_{i,k}$  do
8    $w \leftarrow 0$ ;
9   for  $N_{j,k} \in \{N_{1,k}, \dots, N_{M_k,k}\} \setminus N_{i,k}$  do
10    forall the  $e \in$  Edges connected to  $v$  do
11      forall the  $p \in N_{j,k}$  do
12        if  $p$  is a target of edge  $e$  then
13           $w \leftarrow w + edgeWeight(e)$ ;
14   if  $w < minWeight$  then
15      $minWeight \leftarrow w$ ;
16      $v_{i,k}^{tt} \leftarrow v$ ;
17 return  $v_{i,k}^{tt}$ ;

```

2) *Generating candidate policies for belief modes:* Once we have picked the targets corresponding to each mode, we need to find the control action that can take a mode from its current state to the target state. We generate the candidate trajectory that takes each mode to its target using the RRT* planner [13]. Once an open loop trajectory is computed, we generate a local policy π_i (feedback controller) for the i -th mode, which drives the i -th mode along this trajectory. Let Π be the set of all such policies for the different modes.

3) *Picking the Optimal Policy:* Once we have generated the set Π of candidate policies. We need to evaluate the expected information gain ΔI_i for each policy π_i and pick the optimal

policy π^* that maximizes this information gain. We model this information gain as the discrete change in the number of modes. To compute the expected change in the number of belief modes, we simulate the most-likely belief trajectory, i.e., approximating noisy observations and actions with their most-likely values. We know that one policy may or may not be good for all the modes, i.e., a policy based on one mode may lead to collision for the other modes. Therefore, we need a way of penalizing a candidate policy during the planning stage if it results in collision. We introduce a penalty c_{fail}/T into the information gain where c_{fail} is a fixed value and T is the time step at which the collision takes place during the simulation of belief trajectory under some policy. Thus, policies which result in a collision much further down are penalized less compared to policies that result in immediate collision. The steps to calculate the expected information gain for a candidate policy $\pi_i \in \Pi$ are as follows:

- 1) For every belief mode $m_{j,k} \in b_k$.
 - a) Assume that robot is at $m_{j,k}$.
 - b) Simulate π_i and propagate all the modes.
 - c) Compute information gain $\Delta I_{i,m_{j,k}}$ (reduction in number of modes while factoring in collision cost) for π_i .
- 2) Compute the weighted information gain $\Delta I_i = \sum_{j=1}^{M_k} w_{j,k} \Delta I_{i,m_{j,k}}$.

After computing the expected information gain for each policy, we pick the gain maximizing policy. The computational complexity of picking the optimal policy is $\mathcal{O}(M_k^3 L_{max})$ (where M_k is the number of belief modes and L_{max} is the maximum candidate trajectory length). This is due to the fact that each policy is simulated for each mode for the length of policy, where at every step of policy execution, there are M_k filter updates. Figure 2 depicts the process of picking the optimal candidate trajectory in a simple scenario.

4) *Belief Propagation Using GMM:* We first discuss our decision to use EKF based multi-hypothesis tracking over a particle filtering approach. In practical localization problems, a relatively small number of Gaussian hypothesis are sufficient

for maintaining the posterior over the robot state, secondly the filtering complexity grows linearly in the number of hypothesis and finally due to the complexity of our re-planning step ($\mathcal{O}(M_k^3 L_{max})$), the number of samples required for a particle filter would render re-planning computationally inefficient. Now, we proceed to describe the weight update step which determines how likely each mode is in the belief. In a standard implementation, the weights $w_{i,k}$'s are updated based on the measurement likelihood function as shown in Eq. 3.

$$w_{i,k+1} = w_{i,k} e^{-\frac{1}{2} D_{i,k+1}^2} \quad (3)$$

where $D_{i,k+1}$ is the Mahalanobis distance between the sensor observation and most-likely observation for mode m_i such that

$$D_{i,k+1}^2 = (z_{k+1} - h(\mu_{i,k+1}, 0))^T R_k^{-1} (z_{k+1} - h(\mu_{i,k+1}, 0)). \quad (4)$$

The weights are normalized such that $\sum_{i=1}^{M_k} w_{i,k+1} = 1$. A known issue with EKF-based MHT is that it is unable to process negative information [24]. Negative information refers to the lack of information which one may expect to see and can certainly help in disproving a hypothesis. We now proceed to describe how we factor in negative information in the weight update.

Factoring Negative Information: Depending on the state of the robot, individual hypotheses and data association results, we might have several cases. We discuss this issue in the context of a landmark based measurement model. At time t_{k+1} , let $n_{z_{k+1}}$ be the number of landmarks observed by the robot and $n_{z_{i,k+1}^p}$ be the number of landmarks that we predict to see for m_i where $z_{i,k+1}^p = h(\mu_{i,k+1}, 0)$ is the predicted observation. Then $n_{z_{k+1}} = n_{z_{i,k+1}^p}$ means that the i -th mode expected to see as many landmarks as the robot observed; $n_{z_{k+1}} > n_{z_{i,k+1}^p}$ implies the robot observes more landmarks than predicted for the mode; $n_{z_{k+1}} < n_{z_{i,k+1}^p}$ implies the robot observes less landmarks than predicted for the mode. Also, we can have the number of data associations to be less than the number of predicted or measured observations or both. This means that we may not be able to make a unique association between each predicted and some observed landmark. At time t_{k+1} , we estimate the Mahalanobis distance $D_{i,k+1}$ (Eq. 4) for mode m_i between the predicted and observed landmarks that are matched by the data association module and update weight according to Eq. 3. Then we multiply the updated weight by a factor γ , which models the effect of duration $\beta_{i,k+1}$ for which the robot observes different landmarks than the i -th mode's prediction; and the discrepancy α in the number of data associations. When a belief mode is initialized, we set $\beta_{i,0} = 0$. The weight update procedure is described in Algorithm 3. After each weight update step, we remove modes with negligible contribution to the belief, i.e., when $w_{i,k+1} \leq \delta_w$ where δ_w is a user defined parameter for

minimum weight threshold ¹.

Algorithm 3: GMM Weight Update

```

1 Input:  $w_{i,k}, \mu_{i,k+1}, \beta_{i,k}, \delta t$ 
2 Output:  $w_{i,k+1}, \beta_{i,k+1}$ 
3  $z_{k+1}, n_{z_{k+1}} \leftarrow$  Get sensor observations;
4  $z_{i,k+1}^p, n_{z_{i,k+1}^p} \leftarrow$  Get predicted observations for  $\mu_{i,k+1}$ ;
5  $n_{z_{k+1}} \cap n_{z_{i,k+1}^p} \leftarrow$  Do data association;
6  $w'_{i,k+1} \leftarrow$  Update and normalize weight according to
   likelihood function;
7  $\gamma \leftarrow 1$ ;
8 if  $n_{z_{i,k+1}^p} \neq n_{z_{k+1}}$  or  $n_{z_{i,k+1}^p} \neq n_{z_{k+1}} \cap n_{z_{i,k+1}^p}$  then
9    $\alpha \leftarrow \max(1 + n_{z_{k+1}} - n_{z_{k+1}} \cap n_{z_{i,k+1}^p}, 1 + n_{z_{i,k+1}^p} -$ 
      $n_{z_{k+1}} \cap n_{z_{i,k+1}^p})$ ;
10   $\beta_{i,k+1} \leftarrow \beta_{i,k} + \delta t$ ;
11   $\gamma \leftarrow e^{-\alpha \beta_{i,k+1} 10^{-4}}$ ;
12 else
13   $\beta_{i,k+1} \leftarrow 0$ ;
14  $w_{i,k+1} \leftarrow w'_{i,k+1} \gamma$ ;
15 return  $w_{i,k+1}, \beta_{i,k+1}$ ;

```

C. Analysis

In this section, we show that the receding horizon planner M3P will guarantee that an initial multimodal belief is driven into a unimodal belief in finite time. First, we make the following assumptions:

Assumption 1: Given a multimodal belief b_k , for every mode $m_{i,k}$, the environment allows for the existence of some target state $v_{i,k}^{tt}$ and some homotopy class of paths through which the robot can visit $v_{i,k}^{tt}$, such that if the robot was in mode $m_{i,k}$, it could confirm that m_i is the true hypothesis.

Assumption 2: The map does not change during the execution of the planner.

Proposition 1: Under Assumptions 1 and 2, given any initial multimodal belief $b_0 = \sum_i w_{i,0} m_{i,0}$, the receding horizon planner M3P drives the belief process into a unimodal belief $b_T = m_T \approx \mathcal{N}(\mu_T, \Sigma_T)$ in some finite time T .

Proof: Suppose that the robot is at the initial belief b_0 . Suppose we choose the plan G_{i^*} , i.e., candidate policy for mode i^* , that results in the most information gain as required by the M3P planner. The plan G_{i^*} can be applied to all the modes at least for some finite period of time in the future, since if it cannot be applied, then we immediately know that the robot is not at mode i^* and thus, there is a disambiguation whereby mode i^* is discarded. Once the plan G_{i^*} is executed, there are only 2 possibilities:

1) The robot is able to execute the entire plan G_{i^*} till the end, or 2) the plan becomes unfeasible at some point of its

¹In practical MHT applications, highly similar modes can often be merged to reduce complexity. We refer the reader to [12, 24] for a thorough explanation. From a planning perspective, when two modes are merged, policy execution is halted, and re-planning is triggered. M3P computes candidate policies for the new set of belief modes to disambiguate them.

execution.

In case 1, due to Assumption 1, we will know for sure that the robot was at mode i^* and the belief collapses into a unimodal belief thereby proving the result. In case 2, due to Assumption 2, we know that the robot could not have started at mode i^* and thus, the number of modes is reduced by at least one. After this disambiguation, we restart the process as before and we are assured that at least one of the modes is going to be disambiguated and so on. Thus, it follows given that we had a finite number of modes to start with, the belief eventually converges to a unimodal belief. Further, since each of the disambiguation epochs takes finite time, a finite number of such epochs also takes a finite time, thereby proving the result. ■

Remarks: The above result shows that the M3P algorithm will stabilize the belief process to a unimodal belief under assumptions 1 and 2. In the case that assumption 1 is violated we are either (i) unable to find a target which allows the robot to observe distinctive information (e.g., trivial case of a robot operating in a world with identical infinite corridors) or (ii) we may find such a target but the environment geometry does not allow for any path to visit it (e.g., robot stuck in one of many identical rooms and the doors are closed) or (iii) all homotopy class of paths to visit a target state pass through long regions with no information such that the uncertainty on each mode grows sufficiently high that we cannot make data associations at the target location to disambiguate the multimodal belief. Violations (i) and (ii) refer to degenerate cases that rarely occur in practical motion planning problems. Violation (iii) is currently beyond the scope of this paper and presents an important direction for future research. Assumption 2 (static world) is common in localization literature, though it may be violated in certain practical scenarios. In such cases, if the map is not changing rapidly, one may use sensory observations to incorporate new constraints into the map and trigger replanning.

V. EXPERIMENTAL RESULTS

We present experimental results for a non-holonomic ground robot. The experiments represent two motion planning scenarios wherein the robot is placed randomly at a location in an environment which is identical to other locations in appearance. Thus the initial belief is multimodal, the goal of the experiment is to use the non-Gaussian planner M3P described in Section IV to localize to a unimodal belief. We first describe the system setup to motivate the experiment followed by the results.

A. System Description

We used a low-cost Arduino based differential drive robot as shown in Fig. 3. The robot is equipped with an Odroid U3 computer running ROS on Ubuntu 14.04 and an off-the-shelf Logitech C-310 webcam for sensing. The onboard computer uses a wifi link to communicate with the ground control station. We use an off-the-shelf laptop running Ubuntu 14.04 as the ground control station. The ground station runs the planner

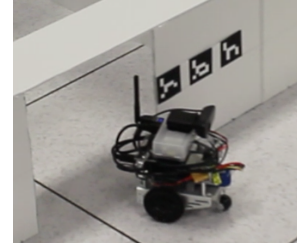


Fig. 3: The robot exiting a room in the maze. The robot has an 11.5 cm wheelbase and measures 18 cm and 15 cm in height and length respectively.

and image processing algorithms while communicating with the robot via wifi.

Motion Model: The kinematics of the robot are represented by a unicycle.

$$x_{k+1} = f(x_k, u_k, w_k) = \begin{pmatrix} x_k + (V_k + n_v)\delta t \cos \theta_k \\ y_k + (V_k + n_v)\delta t \sin \theta_k \\ \theta_k + (\omega_k + n_\omega)\delta t \end{pmatrix}, \quad (5)$$

where $x_k = (x_k, y_k, \theta_k)^T$ describes the robot state (position and yaw angle). $u_k = (V_k, \omega_k)^T$ is the control vector consisting of linear velocity V_k and angular velocity ω_k . We denote the process noise vector by $w_k = (n_v, n_\omega)^T \sim \mathcal{N}(0, \mathbf{Q}_k)$.

Observation Model: Our observation model is based on passive visual beacons/landmarks which can be observed by a monocular camera to measure their relative range, bearing and an associated ID tag. Let the location of the i -th landmark be denoted by ${}^i\mathbf{L}$. The displacement vector ${}^i\mathbf{d}$ from the robot to ${}^i\mathbf{L}$ is given by ${}^i\mathbf{d} = [{}^i d_x, {}^i d_y]^T := {}^i\mathbf{L} - \mathbf{p}$, where $\mathbf{p} = [x, y]^T$ is the position of the robot. Therefore, the observation ${}^i z$ of the i -th landmark can be modeled as,

$${}^i z = {}^i h(x, {}^i v) = [\|{}^i\mathbf{d}\|, \text{atan2}({}^i d_y, {}^i d_x) - \theta]^T + {}^i v, \quad (6)$$

The observation noise is zero-mean Gaussian such that ${}^i v \sim \mathcal{N}(\mathbf{0}, {}^i\mathbf{R})$ where ${}^i\mathbf{R} = \text{diag}((\eta_r \|{}^i\mathbf{d}\| + \sigma_r^r)^2, (\eta_\theta \|{}^i\mathbf{d}\| + \sigma_b^\theta)^2)$. The quality of sensor reading decreases as the robot gets farther from the landmarks. The parameters η_r and η_θ determine this dependency, and σ_b^r and σ_b^θ are the bias standard deviations.

B. Scenario

We constructed a symmetrical maze that has 8 identical rooms (R1-8) as shown in Fig. 4. Augmented reality (AR) markers were placed on the walls which act as the landmarks detected by the vision-based sensing system on the robot [9].

Parameter	Value
δ_w	0.01
c_{fail}	10^6
RHC horizon	60 secs

TABLE I: List of experiment parameters.

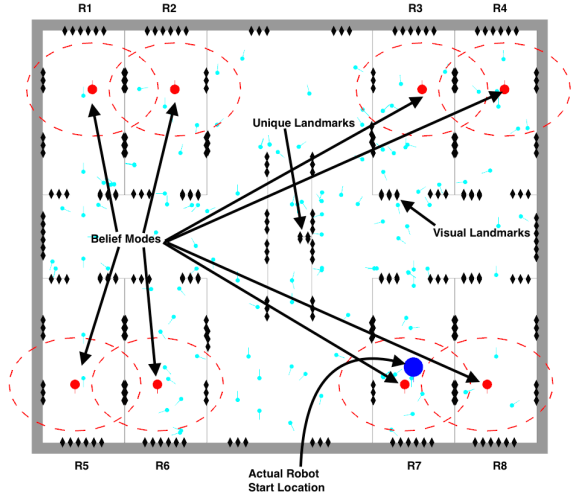


Fig. 4: The environment and belief at the start of first run. The robot is placed in room R7 (blue disk). The initial sampling leads to 8 belief modes, one in each room. The black diamonds mark the locations of augmented reality markers in the environment. The unique landmarks are placed inside the narrow passage, such that if the robot enters the passage from either side, it sees distinctive information.

Each marker has a signature (an ID number), thus when the robot sees a landmark, it can measure the range, bearing as well as its ID number. To create ambiguity in the data association, we placed multiple AR markers with the same ID number in different parts of the environment. For example, one of the symmetries in our experiment is the inside of each room. Each room in the maze appears identical to the robot as markers with the same appearance are placed on each room's walls with an identical layout. Thus, if the robot is placed in a location with markers similar to another part of the environment, the data associations lead the robot to believe it could be in one of these many locations, which leads to a multimodal belief on the state. We also place four unique landmarks in a narrow passage in the center of maze as marked in Fig. 4. To successfully localize, the robot must visit this location in order to converge to its true belief.

The robot is initially placed in room R7 and is not given any prior information of its state. We assume a known map. To estimate b_0 , we uniformly sample the configuration space and set these samples as the means $\mu_{i,k}$ of the modes of the Gaussian mixture components and assign identical covariance and uniform weight to each mode. After this, the robot remains stationary and the sensory measurements are used to update the belief state and remove the unlikely modes. Unlikely modes whose weight falls below a predesignated threshold δ_w are rejected. This process of elimination continues until we converge to a fixed number of modes. Figure 4 shows the initial belief. The robot plans its first set of candidate actions as shown in Fig. 5(a). After the candidates are evaluated, the policy based on mode m_5 in room R5 is chosen and executed. As the robot turns, it sees a landmark on the wall outside

the room (shown in Fig. 5(b)). This causes mode m_4 to be deleted. Immediately, replanning is triggered and a new set of candidate trajectories is created. In successive steps, we see that first modes m_3 and m_5 are deleted and then after the next two replanning steps, modes m_8 , m_1 and m_6 are deleted. We notice that the robot does not move till only the 2 most-likely modes are remaining. The reason for this is that seeing the marker on the outside wall has the effect of successively lowering the weights of the unlikely modes due to filtering. As the mode weights fall below the threshold, they are deleted, which triggers the replanning condition. Once the belief has converged to the two most-likely modes m_2, m_7 (as expected by the symmetry) a new set of candidate policies is created and the policy based on mode m_2 is chosen. This policy leads the modes out of the rooms, and towards the narrow passage. Figure 5(c) shows both belief modes executing the policy based on mode m_2 . While executing this policy, replanning is triggered as the robot exceeds maximum horizon for policy execution. The final policy drives the robot into the narrow passage and the unique landmarks are observed (Fig. 5(d)) which leads the belief to converge to the true belief.

Due to paucity of space we only present one experiment here, a supplementary video is provided that clearly depicts every stage of both our experiments.

C. Discussion

Our approach results in an intuitive behavior which guides the robot to seek disambiguating information such that it can sequentially reject the incorrect hypothesis about its state. The candidate trajectories are regenerated every time a Gaussian mode is rejected or a constraint violation is foreseen. The time to re-plan reduces drastically as the number of modes reduce. Thus, the first few actions are the hardest which is to be expected as we start off with a large number of hypotheses. Finally, the planner is able to localize the robot safely. In [8], the authors showed that random motion is inefficient and generally incapable of localizing a robot within reasonable time horizons especially in cases with symmetry (e.g., office environments with long corridors and similar rooms). In [12] the authors consider the robot localized when one of the modes gets a weight ≥ 0.8 , in contrast our approach is more conservative in that we only consider the robot localized when a mode has weight ≥ 0.99 . We can afford to be more conservative as our localization strategy actively seeks disambiguating information using prior map knowledge as opposed to a heuristic based strategy. While our experiment acts as a proof of concept, there are certain phenomenon such as cases where the belief modes split into child modes, or dynamic environments which were not covered and will be addressed in future work.

VI. CONCLUSION

In this work, we studied the problem of motion planning for a mobile robot when the underlying belief state is non-Gaussian in nature. Our main contribution in this work is a planner M3P that generates a sequentially disambiguating policy, which leads the belief to converge to a unimodal

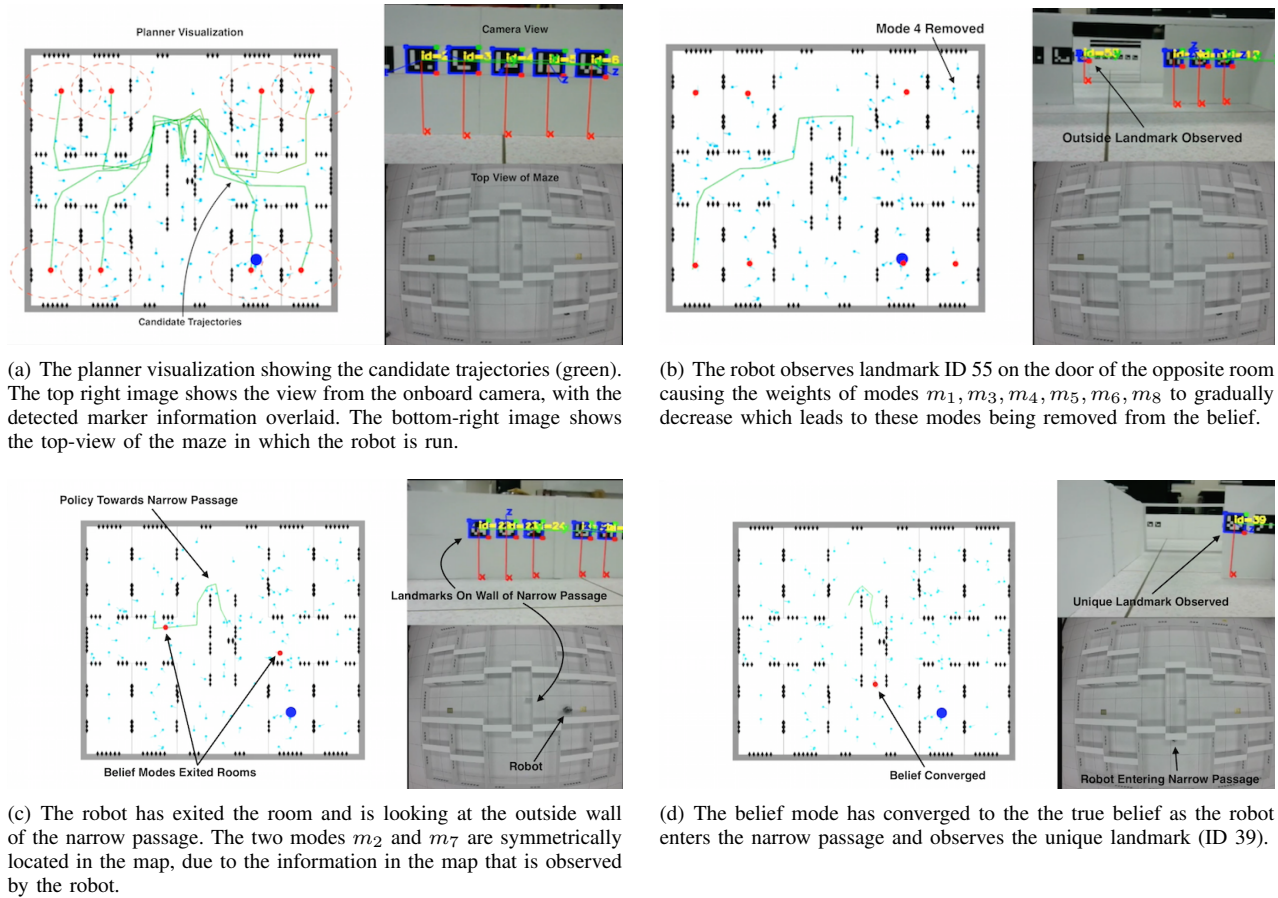


Fig. 5: Snapshots of the experiment at different times.

Gaussian. We are able to show in practice that the robot is able to recover from a kidnapped state and localize in environments that present ambiguous data associations. Compared to previous experimental work, we take an active approach to candidate policy generation and selection. However, a key limitation arises in our policy selection step due to algorithmic complexity. This needs to be addressed in future work such that the algorithm can scale well to larger maps which may result in a greater number of hypotheses. In future work, experiments will be extended to full-scale problems (e.g., office like environments where symmetries are known to cause ambiguity in the belief) and more drastic localization failures (e.g., when a well-localized robot is kidnapped to a random location, the a priori information tends to mislead the robot). Finally, there may be tasks which are feasible with a multimodal distribution on the belief. Such cases present an interesting area for future motion planning research.

VII. ACKNOWLEDGMENTS

We would like to thank Shiva Gopalan for helping us in programming the robot and assembling the maze for the experimental work in this paper.

BIBLIOGRAPHY

- [1] Aliakbar Agha-mohammadi, Saurav Agarwal, Aditya Mahadevan, Suman Chakravorty, Daniel Tomkins, Jory Denny, and Nancy Amato. Robust online belief space planning in changing environments: Application to physical mobile robots. In *IEEE Int. Conf. Robot. Autom. (ICRA)*, Hong Kong, China, 2014.
- [2] Aliakbar Agha-mohammadi, Suman Chakravorty, and Nancy Amato. FIRM: Sampling-based feedback motion planning under motion uncertainty and imperfect measurements. *International Journal of Robotics Research*, 33(2):268–304, 2014.
- [3] Dimitri Bertsekas. *Dynamic Programming and Optimal Control: 3rd Ed.* Athena Scientific, 2007.
- [4] Adam Bry and Nicholas Roy. Rapidly-exploring random belief trees for motion planning under uncertainty. In *ICRA*, pages 723–730, 2011.
- [5] S. Chakravorty and R. Scott Erwin. Information space receding horizon control. In *IEEE Symposium on Adaptive Dynamic Programming And Reinforcement Learning (ADPRL)*, April 2011.
- [6] Pratik Chaudhari, Sertac Karaman, David Hsu, and Emilio Frazzoli. Sampling-based algorithms for continuous-time POMDPs. In *the American Control Conference (ACC)*, Washington DC, 2013.
- [7] Gregory Dudek, Kathleen Romanik, and Sue Whitesides. Localizing a robot with minimum travel. *SIAM Journal on Computing*, 27(2):583–604, 1998. doi: 10.1137/S0097539794279201.
- [8] Dieter Fox, Wolfram Burgard, and Sebastian Thrun. Active markov localization for mobile robots. *Robotics and Autonomous Systems*, 25(34):195 – 207, 1998. ISSN 0921-8890. doi: [http://dx.doi.org/10.1016/S0921-8890\(98\)00049-9](http://dx.doi.org/10.1016/S0921-8890(98)00049-9). Autonomous Mobile Robots.
- [9] S. Garrido-Jurado, R. Muñoz-Salinas, F.J. Madrid-Cuevas, and M.J. Marn-Jimnez. Automatic generation and detection of highly reliable

- fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280 – 2292, 2014. ISSN 0031-3203. doi: <http://dx.doi.org/10.1016/j.patcog.2014.01.005>.
- [10] A. Gasparri, S. Panziera, F. Pascucci, and G. Ulivi. A hybrid active global localisation algorithm for mobile robots. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 3148–3153, April 2007. doi: 10.1109/ROBOT.2007.363958.
 - [11] R. He, E. Brunskill, and N. Roy. Efficient planning under uncertainty with macro-actions. *Journal of Artificial Intelligence Research*, 40:523–570, February 2011.
 - [12] P. Jensfelt and S. Kristensen. Active global localization for a mobile robot using multiple hypothesis tracking. *Robotics and Automation, IEEE Transactions on*, 17(5):748–760, Oct 2001. ISSN 1042-296X.
 - [13] Sertac Karaman and Emilio Frazzoli. Sampling-based algorithms for optimal motion planning. *International Journal of Robotics Research*, 30(7):846–894, June 2011.
 - [14] L.E. Kavraki, P. Svestka, J.C. Latombe, and M. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Transactions on Robotics and Automation*, 12(4):566–580, 1996.
 - [15] H. Kurniawati, T. Bandyopadhyay, and N.M. Patrikalakis. Global motion planning under uncertain motion, sensing, and environment map. *Autonomous Robots*, 33(3):255–272, 2012.
 - [16] V. Pilania and K. Gupta. A localization aware sampling strategy for motion planning under uncertainty. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 6093–6099, Sept 2015. doi: 10.1109/IROS.2015.7354245.
 - [17] R. Platt. Convex receding horizon control in non-Gaussian belief space. In *Workshop on the Algorithmic Foundations of Robotics (WAFR)*, 2012.
 - [18] R. Platt, L. Kaelbling, T. Lozano-Perez, , and R. Tedrake. Efficient planning in non-Gaussian belief spaces and its application to robot grasping. In *Proc. of International Symposium of Robotics Research (ISRR)*, 2011.
 - [19] Robert Platt, Russ Tedrake, Leslie Kaelbling, and Tomas Lozano-Perez. Belief space planning assuming maximum likelihood observations. In *Proceedings of Robotics: Science and Systems (RSS)*, June 2010.
 - [20] Robert Platt, Leslie Kaelbling, Tomas Lozano-Perez, and Russ Tedrake. Non-gaussian belief space planning: Correctness and complexity. In *ICRA*, 2012.
 - [21] Sam Prentice and Nicholas Roy. The belief roadmap: Efficient planning in belief space by factoring the covariance. *International Journal of Robotics Research*, 28(11-12), October 2009.
 - [22] J. Reuter. Mobile robot self-localization using pdab. In *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*, volume 4, pages 3512–3518 vol.4, 2000.
 - [23] Stergios I Roumeliotis and George A Bekey. Bayesian estimation and kalman filtering: A unified framework for mobile robot localization. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 3, pages 2985–2992. IEEE, 2000.
 - [24] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*. MIT press, 2005.
 - [25] Jur van den Berg, Pieter Abbeel, and Ken Goldberg. LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information. In *Proceedings of Robotics: Science and Systems (RSS)*, June 2010.